

REMARKS

Claims 1-25 were pending in this application.

Claims 1-25 have been rejected.

Claims 15-20 and 25 have been amended as shown above.

Claims 1-25 remain pending in this case.

Reconsideration and full allowance of Claims 1-25 are respectfully requested.

I. REJECTION UNDER 35 U.S.C. § 103

The Office Action rejects Claims 1-25 under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,630,066 by Gosling (“*Gosling*”) in view of Jones et al., “The Spineless Tagless G-machine” (“*Jones*”). This rejection is respectfully traversed.

In *ex parte* examination of patent applications, the Patent Office bears the burden of establishing a *prima facie* case of obviousness. MPEP § 2142; *In re Fritch*, 972 F.2d 1260, 1262, 23 U.S.P.Q.2d 1780, 1783 (Fed. Cir. 1992). The initial burden of establishing a *prima facie* basis to deny patentability to a claimed invention is always upon the Patent Office. MPEP § 2142; *In re Oetiker*, 977 F.2d 1443, 1445, 24 U.S.P.Q.2d 1443, 1444 (Fed. Cir. 1992); *In re Piasecki*, 745 F.2d 1468, 1472, 223 U.S.P.Q. 785, 788 (Fed. Cir. 1984). Only when a *prima facie* case of obviousness is established does the burden shift to the applicant to produce evidence of nonobviousness. MPEP § 2142; *In re Oetiker*, 977 F.2d 1443, 1445, 24 U.S.P.Q.2d 1443, 1444 (Fed. Cir. 1992); *In re Rijckaert*, 9 F.3d 1531, 1532, 28 U.S.P.Q.2d 1955, 1956 (Fed. Cir. 1993). If the Patent Office does not produce a *prima facie* case of unpatentability, then

without more the applicant is entitled to grant of a patent. *In re Oetiker*, 977 F.2d 1443, 1445, 24 U.S.P.Q.2d 1443, 1444 (Fed. Cir. 1992); *In re Grabiak*, 769 F.2d 729, 733, 226 U.S.P.Q. 870, 873 (Fed. Cir. 1985).

A *prima facie* case of obviousness is established when the teachings of the prior art itself suggest the claimed subject matter to a person of ordinary skill in the art. *In re Bell*, 991 F.2d 781, 783, 26 U.S.P.Q.2d 1529, 1531 (Fed. Cir. 1993). To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed invention and the reasonable expectation of success must both be found in the prior art, and not based on applicant's disclosure. MPEP § 2142.

Gosling recites a "class loader" that downloads objects and object viewers from a remote computer node. (*Abstract*). If a viewer is downloaded, the class loader invokes a "program verification procedure" (also referred to as a "program verifier") to verify the integrity of the viewer. (*Abstract*).

The program verifier (element 240) of *Gosling* uses a virtual stack (element 344) to check the integrity of the viewer before the viewer is actually executed. (*Col. 9, Lines 52-57*). The virtual stack stores "data type information," which identifies the data type of each piece of data

that would be placed in an operand stack by the viewer when it is executed. (*Col. 9, Lines 61-63*).

As an example, if an instruction in the viewer would push an integer value onto the operand stack during execution, the program verifier places an "I" value (indicating integer) onto the virtual stack. (*Col. 10, Lines 1-16*). If another instruction in the viewer would attempt to pop an integer value off of the operand stack during execution, the program verifier determines whether the first character in the virtual stack is an "I" value. (*Col. 13, Lines 19-24*). If the popped data value is of the wrong data type, such as when an instruction expects an integer value and pops a floating point value from the stack, the program verifier of *Gosling* generates an error. (*Col. 13, Lines 29-36*).

It is clear that the program verifier of *Gosling* in no way attempts to verify whether data stored in a storage device has deviated from an "initial state" of that data. Instead, the program verifier of *Gosling* simply uses the contents in the virtual stack to ensure that instructions in a program, if executed, would have access to the correct type of data in the operand stack. *Gosling* in no way uses the virtual stack to inspect a "quantity of information" so as to "identify any deviation from the initial state and thereby detect corruption associated with the stack in the storage device" as recited in the claims.

Jones is not cited by the Office Action as disclosing, teaching, or suggesting the inspection of a "quantity of information" so as to "identify any deviation from the initial state and thereby detect corruption associated with the stack in the storage device." As a result, the

Office Action fails to show that the proposed *Gosling-Jones* combination discloses, teaches, or suggests these elements in the claims.

Moreover, the Office Action asserts that *Jones* discloses inserting a “quantity of information adjacent to the stack in the storage device.” However, *Jones* contains no such recitation. Instead, *Jones* discloses the use of a single stack to store both pointers and non-pointers. (*Page 195, Left column, Second paragraph*). *Jones* notes that some mechanism is needed to distinguish the pointers and non-pointers in the single stack. (*Page 195, Left column, Third paragraph*). *Jones* then notes several possible approaches.

One approach involves tagging each element in the stack with a bit to identify whether it is a pointer. (*Page 195, Left column, Third paragraph*). This could be done either by “stealing” one of the 32 bits in a machine word or stacking the tags. (*Page 195, Left column, Third paragraph*). The Applicant respectfully notes that either of these approaches involves using the contents of the stack itself. In other words, neither of these approaches involves inserting data “adjacent to the stack” as recited in the claims.

Another approach to identifying pointers and non-pointers in the stack is to insert a bit-mask “immediately preceding the code pointed to by [a] return address.” (*Page 195, Right column, First paragraph*). As noted in *Jones*, the return address is contained in the stack. (*Page 195, Left column, Last paragraph*). As clearly shown in Figure 5 of *Jones*, a return address contained in the stack points to a “code block,” and the bit-mask is inserted directly before the code block. (*Figure 5*). The Applicant respectfully notes that the bit-mask used in *Jones* is placed immediately before a block of code. The bit-mask is not placed immediately before or

after the stack. *Jones* lacks any mention of placing the bit-mask “adjacent to the stack” as recited in the claims.

For these reasons, *Jones* fails to disclose, teach, or suggest inserting a “quantity of information adjacent to the stack in the storage device” as recited in the claims. The Office Action acknowledges that *Gosling* does not disclose this element of the claims. As a result, the Office Action fails to show that the proposed *Gosling-Jones* combination discloses, teaches, or suggests inserting a “quantity of information adjacent to the stack in the storage device” as recited in the claims.

Accordingly, the Applicant respectfully requests withdrawal of the § 103(a) rejection and full allowance of Claims 1-25.

II. CONCLUSION

For the reasons given above, the Applicant respectfully requests reconsideration and full allowance of all pending claims and that this application be passed to issue.

SUMMARY

If any outstanding issues remain, or if the Examiner has any further suggestions for expediting allowance of this application, the Applicants respectfully invite the Examiner to contact the undersigned at the telephone number indicated below or at *jmockler@davismunck.com*.

The Applicants have included a Petition for Extension of Time and the appropriate fee for a one (1) month extension of time. No additional fees are believed to be necessary. However, in the event that any additional fees are required for the prosecution of this application, please charge any necessary fees to Deposit Account No. 50-0208.

Respectfully submitted,

DAVIS MUNCK, P.C.

Date:

28 July 2003

P.O. Drawer 800889
Dallas, Texas 75380
Phone: (972) 628-3600
Fax: (972) 628-3616
E-mail: *jmockler@davismunck.com*

John T. Mockler

John T. Mockler
Registration No. 39,775